

INSTITUT DE MATHÉMATIQUES D'ORSAY  
UNIVERSITÉ PARIS-SACLAY

---

# Rapport sur les algorithmes d'approximation et de méta-heuristique pour le problème de l'arbre de Steiner

Éric Aubinais, Farid Najar  
Master Mathématiques de l'Intelligence Artificielle

---

# Table des matières

0.1	Contexte . . . . .	1
0.2	Modélisation . . . . .	1
0.3	Complexité . . . . .	1
0.4	Algorithme et taux d'approximation . . . . .	1
0.5	Performances . . . . .	1
<b>1</b>	<b>Problème 2</b>	<b>2</b>
1.1	Contexte . . . . .	2
1.2	Modélisation . . . . .	2
1.3	Complexité . . . . .	2
1.4	Algorithme et taux d'approximation . . . . .	2
1.5	Performances . . . . .	2
<b>2</b>	<b>Problème 3</b>	<b>3</b>
2.1	Contexte . . . . .	3
2.2	Modélisation . . . . .	3
2.3	Complexité . . . . .	3
2.4	Algorithme et taux d'approximation . . . . .	3
2.5	Performances . . . . .	3

## 0.1 Contexte

Imaginons que nous avons un réseau avec une source et plusieurs destinations. Nous voulons avoir des chemins depuis la source vers les destinations sans être coupé par d'autres chemins. Dit autrement, nous voulons un graphe sans cycles, i.e. un arbre, en partant de la source et en ayant les destinations comme feuilles. De plus, pour aller d'un point à l'autre faut payer un coût. Le but est de trouver l'arbre qui vérifie les conditions énoncées de coût minimal.

## 0.2 Modélisation

Nous pouvons modéliser le réseau par un graphe connexe  $G = (V, E)$  avec  $V$  l'ensemble des sommets et  $E$  l'ensembles des arrêtes. On introduit aussi une fonction  $w : E \rightarrow \mathbb{N}$  qui à chaque arrête  $e \in E$  attribut un poids  $w(e)$ . Soit  $T$  l'ensemble des "terminaux" qui sont tout simplement les destinations que nous avons énoncé. On remarque que la source peut être considérée comme un terminal. En effet, comme notre solution est un arbre, la source est aussi une feuille. Alors notre problème est de trouver un arbre  $A = (V', E')$  (graphe connexe sans cycles), tel que,  $T \subseteq V'$  et on veut minimiser  $\sum_{e \in E'} w(e)$ .

## 0.3 Complexité

## 0.4 Algorithme d'approximation et taux d'approximation

## 0.5 Méta-heuristique

## 0.6 Résultats et performances